



(REPLACEMENT SHEET)

1/14

HURAS ET AL.
YOR920030458US1 (GHZ)

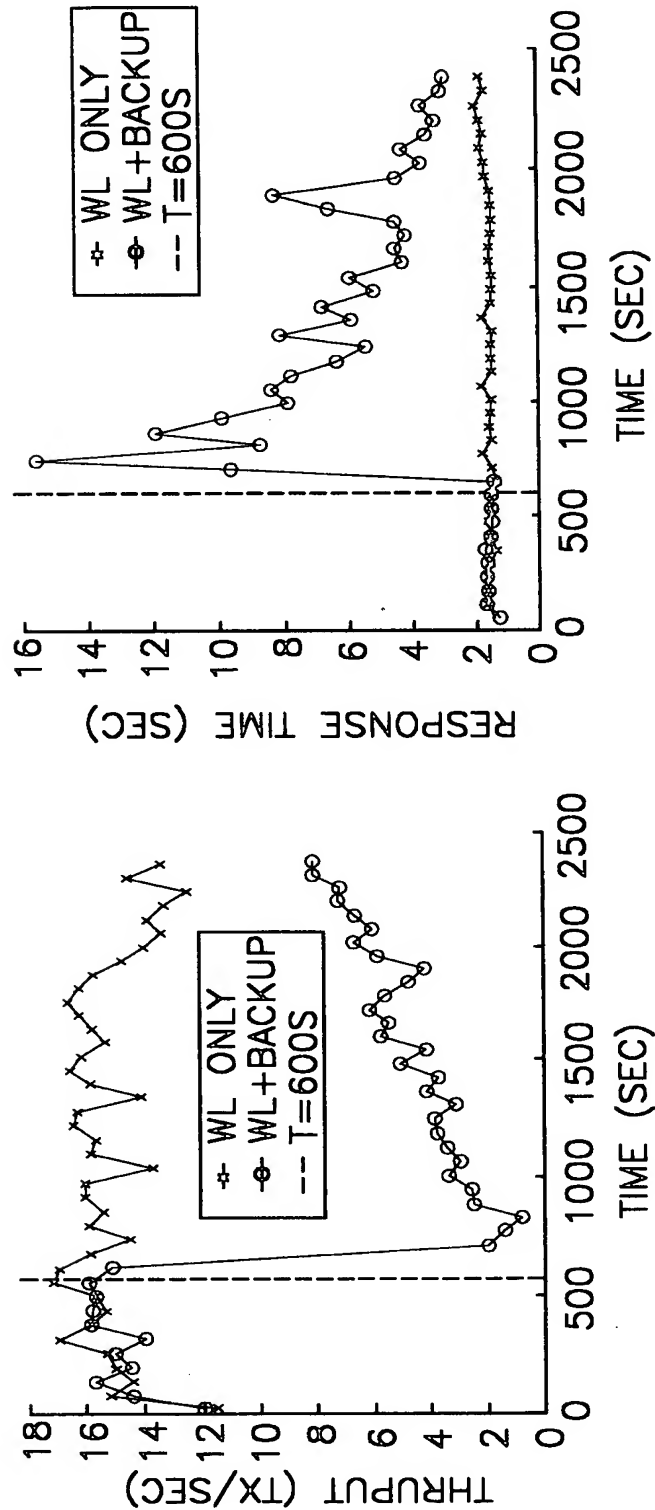


FIG. 1

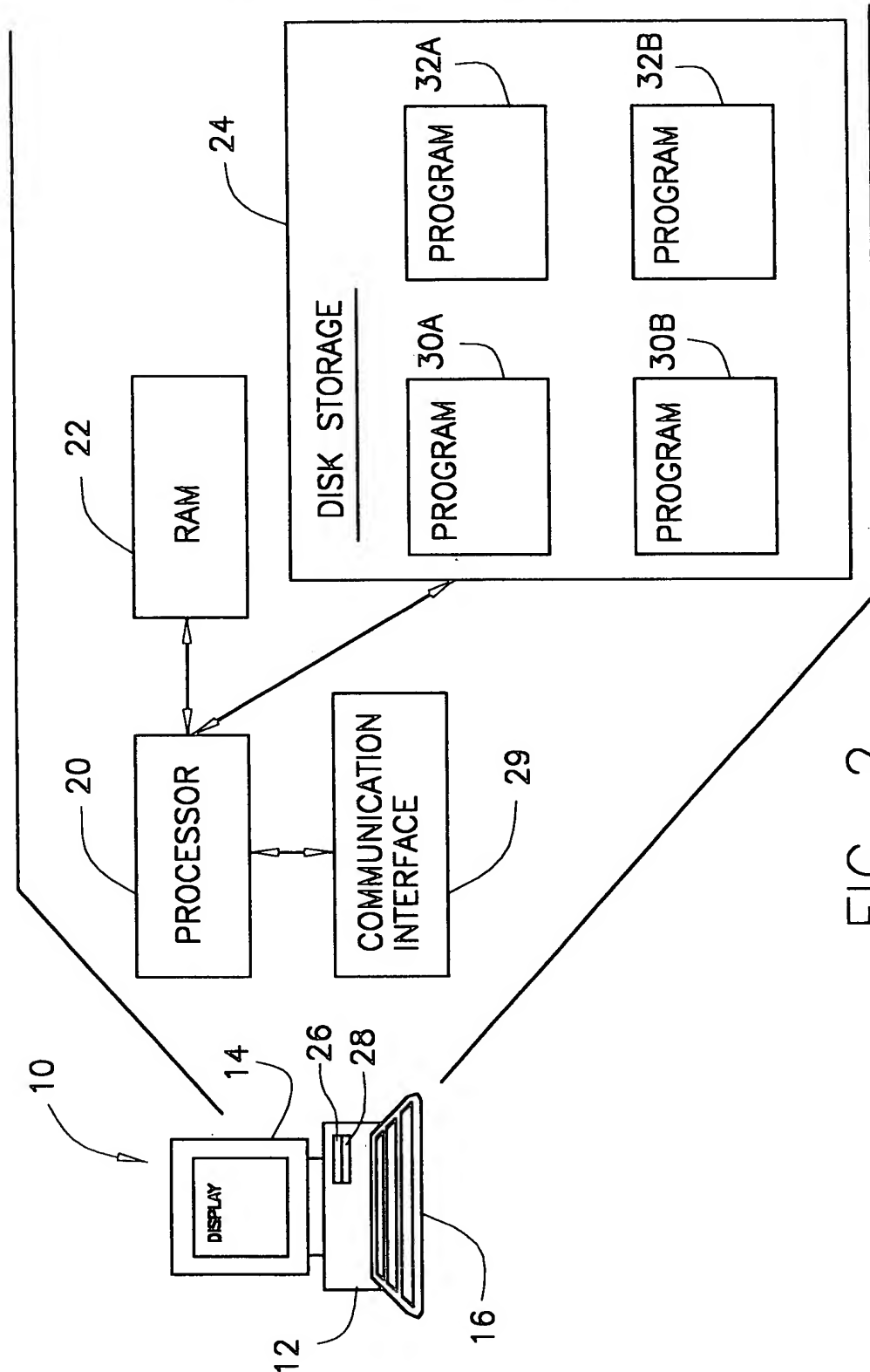


FIG. 2

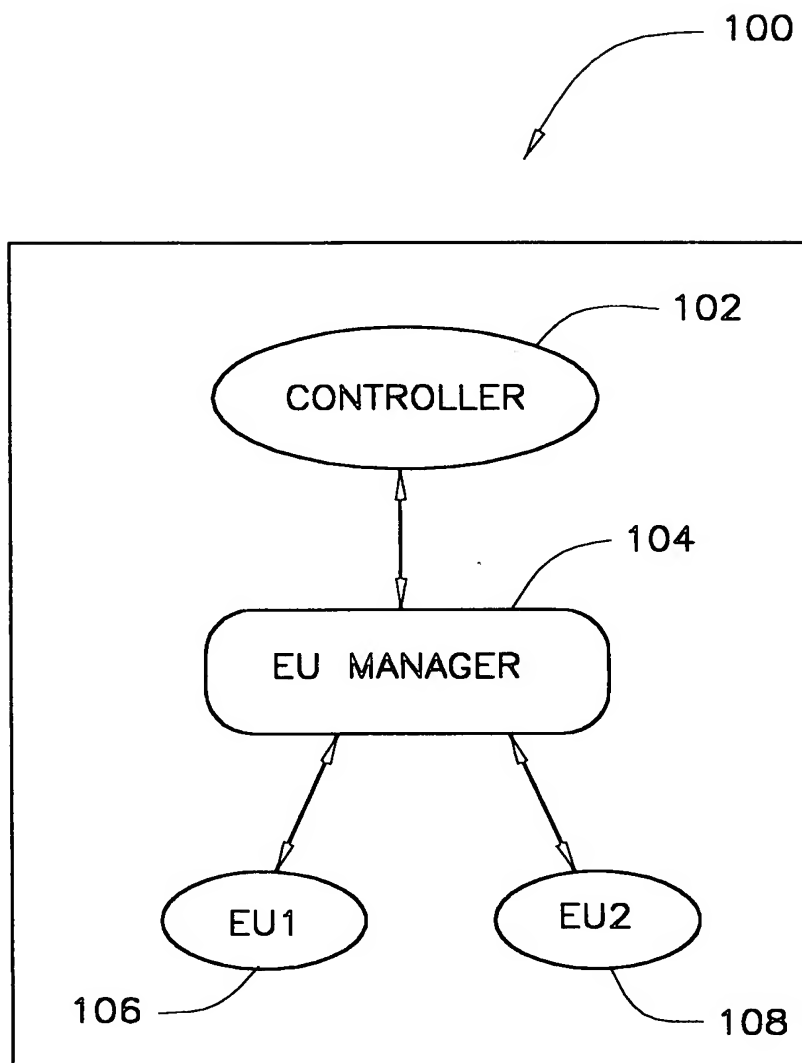


FIG. 3

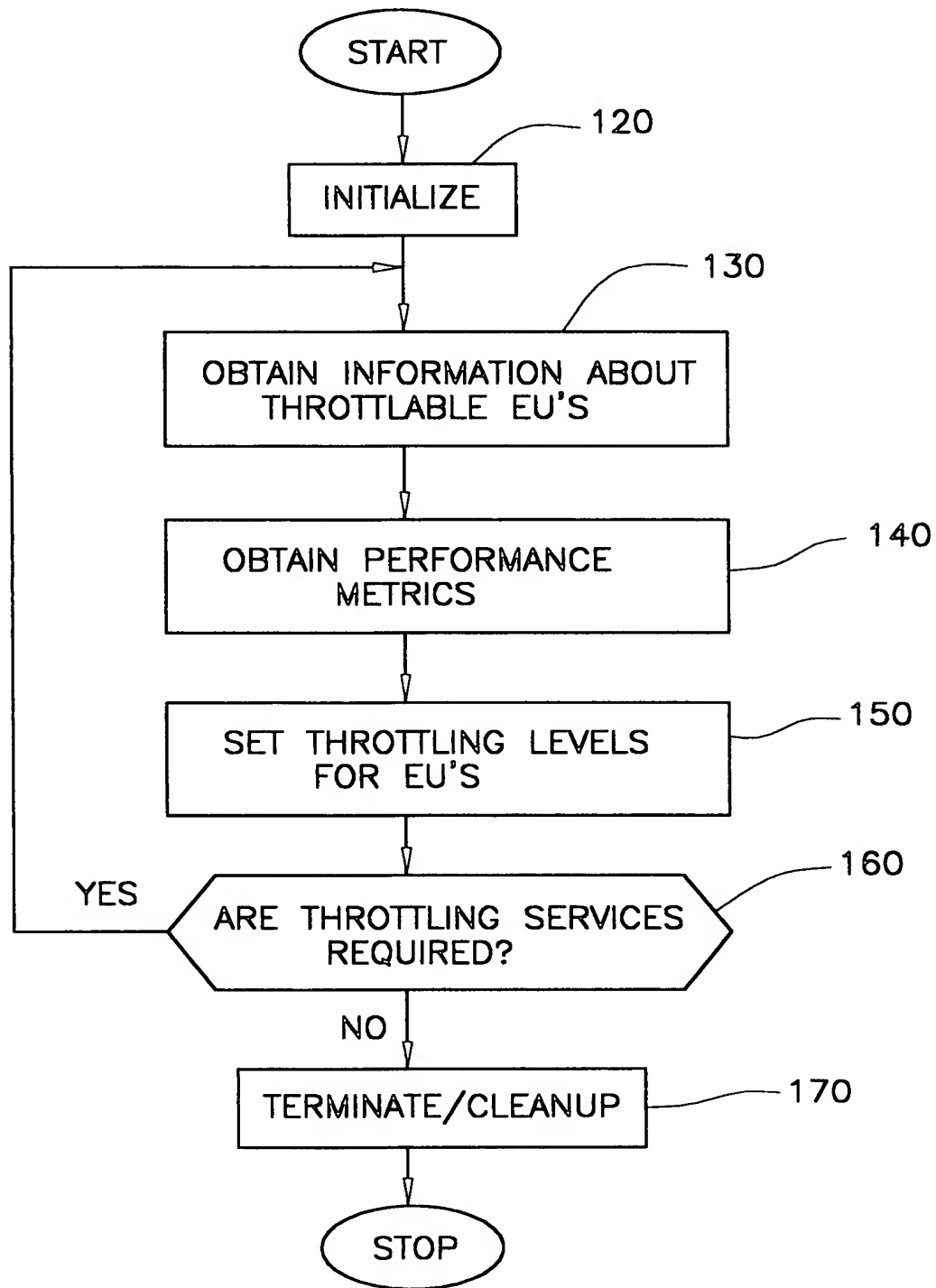


FIG. 4

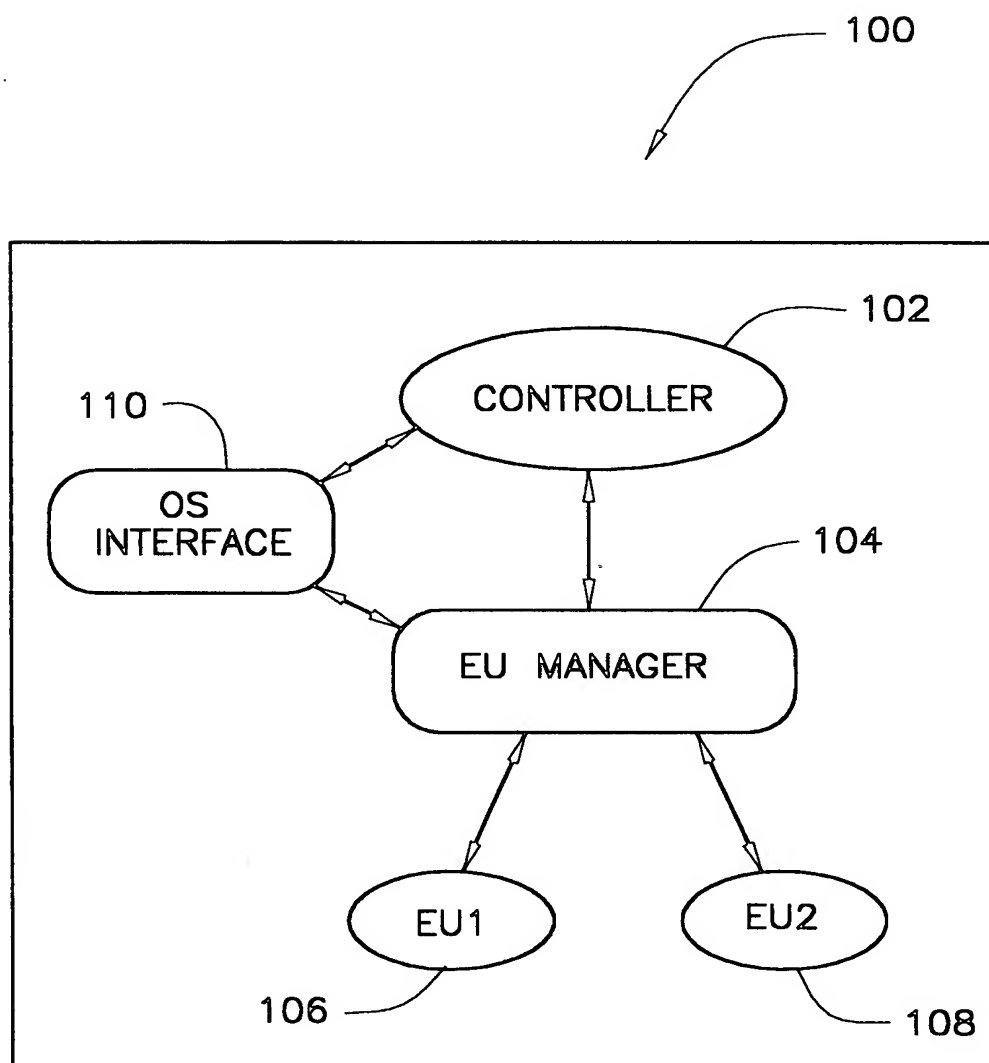


FIG. 5

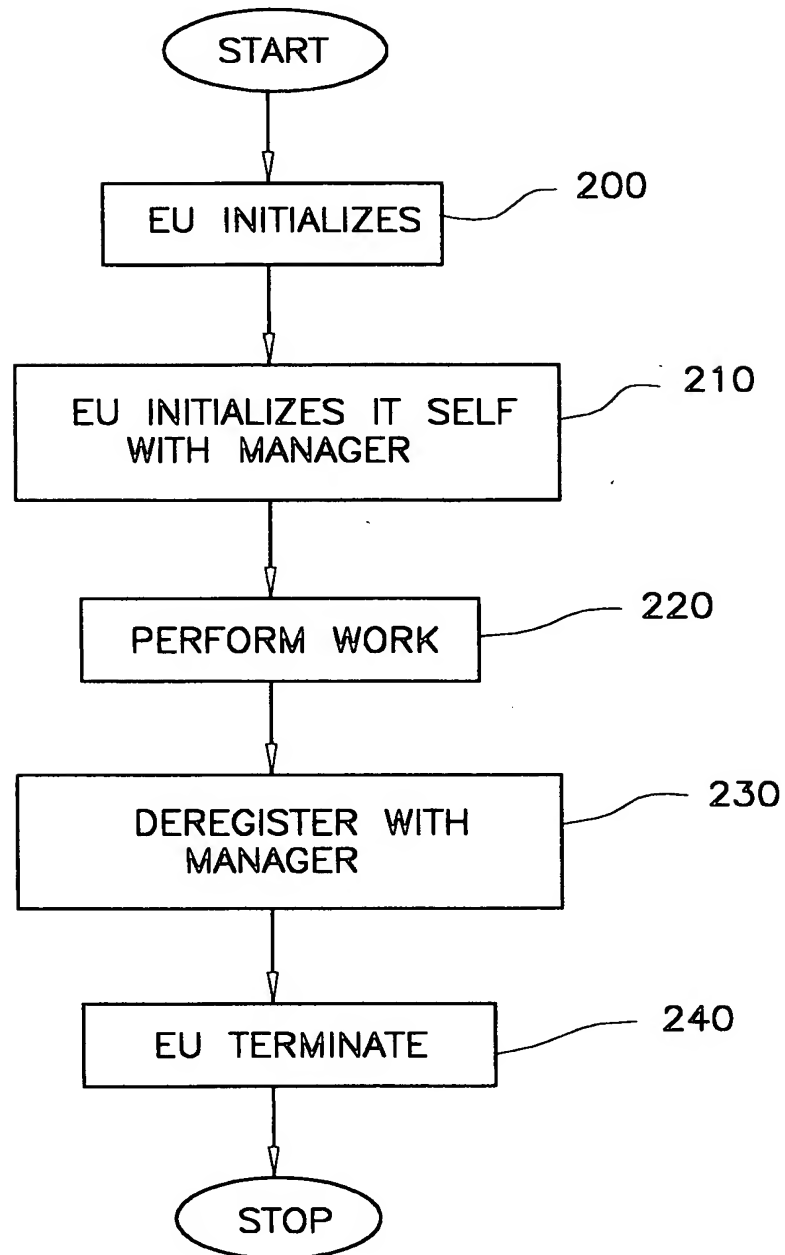


FIG. 6

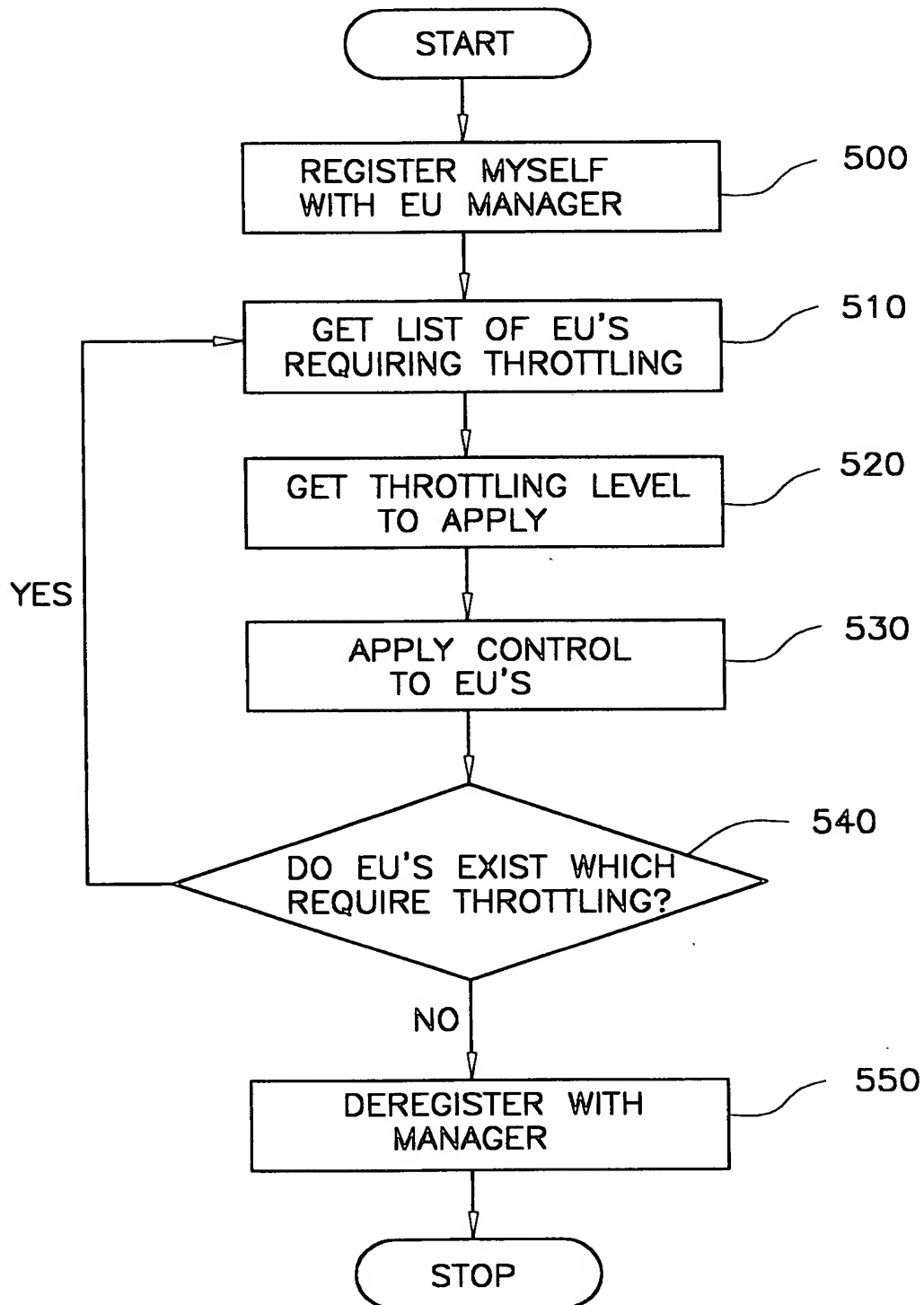


FIG. 6B

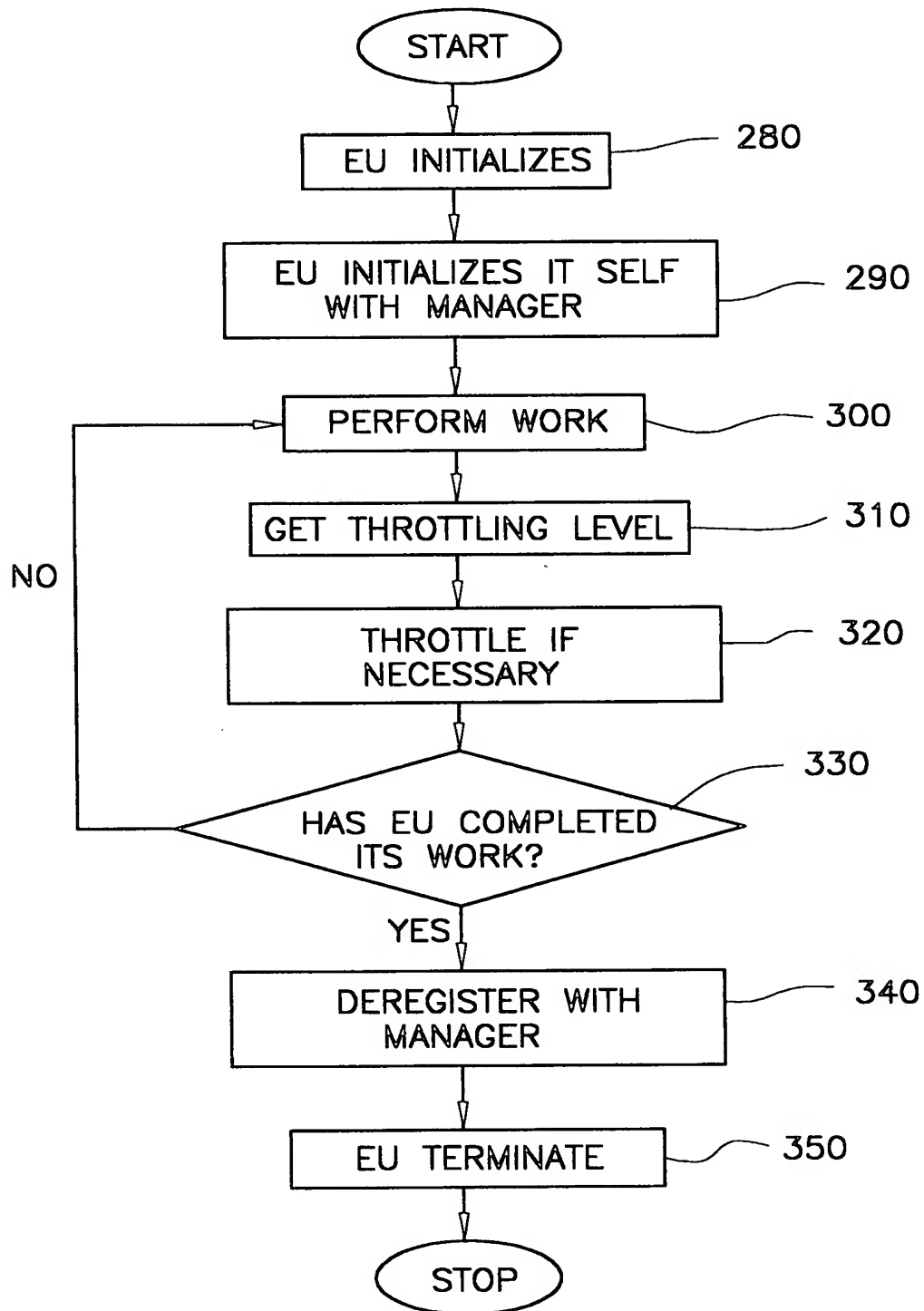


FIG. 7

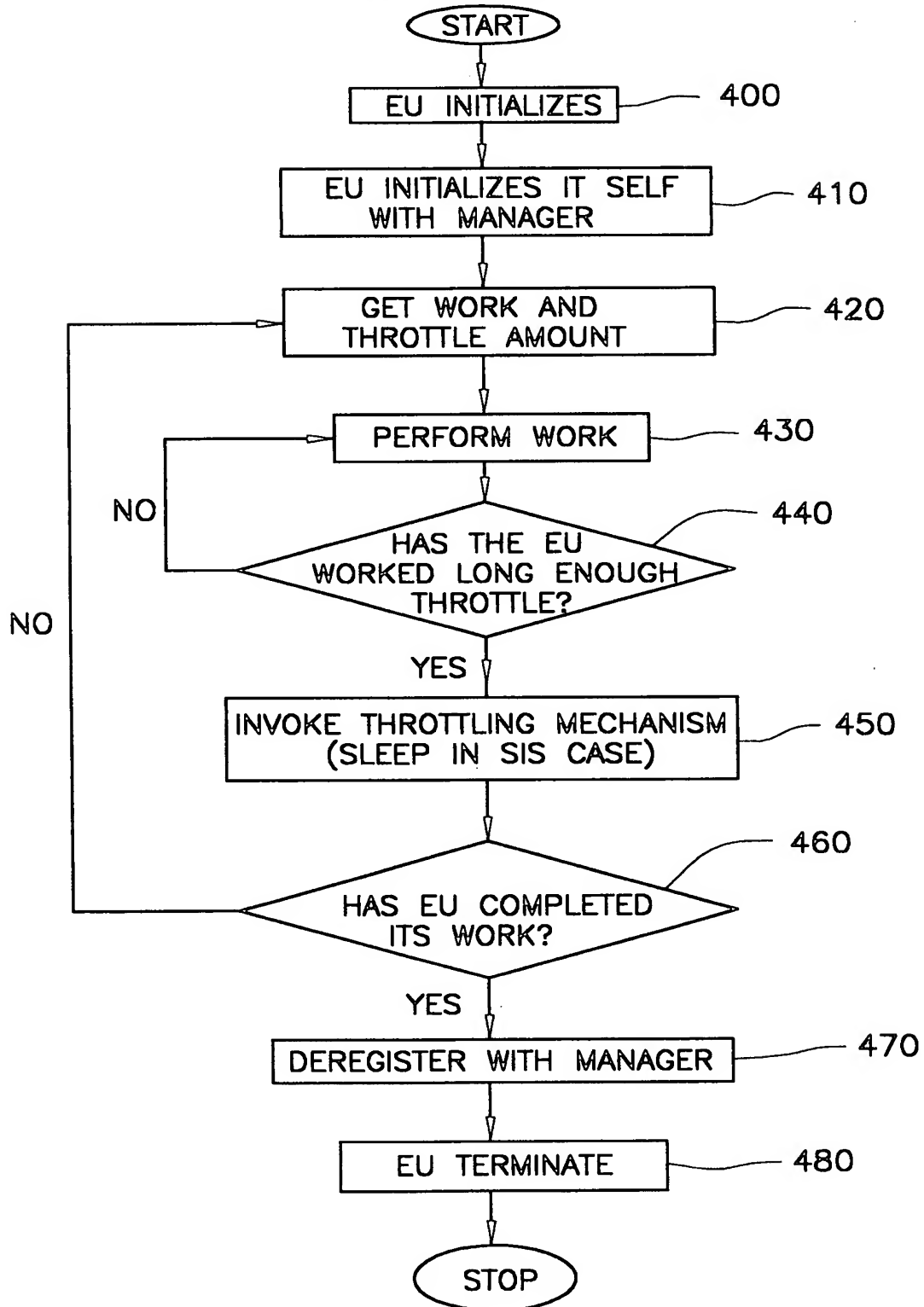


FIG. 8

(REPLACEMENT SHEET)
10/14
HURAS ET AL.
YOR920030458US1 (GHZ)

```
COMPONENT ThrottleAgent
BEGIN
EUManager->RegisterThrottleAgent();
While (EUManager->IsThrottlingRequired())
BEGIN
EUList=EUManager->GetListOfEUs(self);
ThrottlingLevels=EUManager->GetThrottlingLevel(EUList);
...apply ThrottlingLevel to EUs In EUList....
END
EUManager->DeregisterThrottleAgent() ;
END
```

FIG. 9

(REPLACEMENT SHEET)
11/14
HURAS ET AL.
YOR920030458US1 (GHZ)

```
COMPONENT EU
BEGIN
...Initialization...
EUManager->RegisterEU(...args...);

While(NOT Done)
BEGIN
...Do some work...
ThrottlingLevel=EUManager->GetThrottlingLevel(self);
Throttle(ThrottlingLevel);
END

...Termination...
EUManager->DeRegisterEU();
END
```

FIG. 10

(REPLACEMENT SHEET)
12/14
HURAS ET AL.
YOR920030458US1 (GHZ)

```
FUNCTION Utility()  
BEGIN  
    WHILE (NOT done)  
    BEGIN  
        ...do some work...  
        Throttle If Needed()  
    END  
END
```

(a) Inserting SIS point

```
FUNCTION Throttle If Needed()  
BEGIN  
    (workTime, sleepTime) = GetThrottlingLevel() ;  
    timeWorked = Now() - workStart ;  
    IF (timeWorked > workTime)  
        SLEEP ( sleepTime ) ;  
        workStart = Now() ;  
    ENDIF  
END
```

(b) SIS implementation

(REPLACEMENT SHEET)
13/14
HURAS ET AL.
YOR920030458US1 (GHZ)

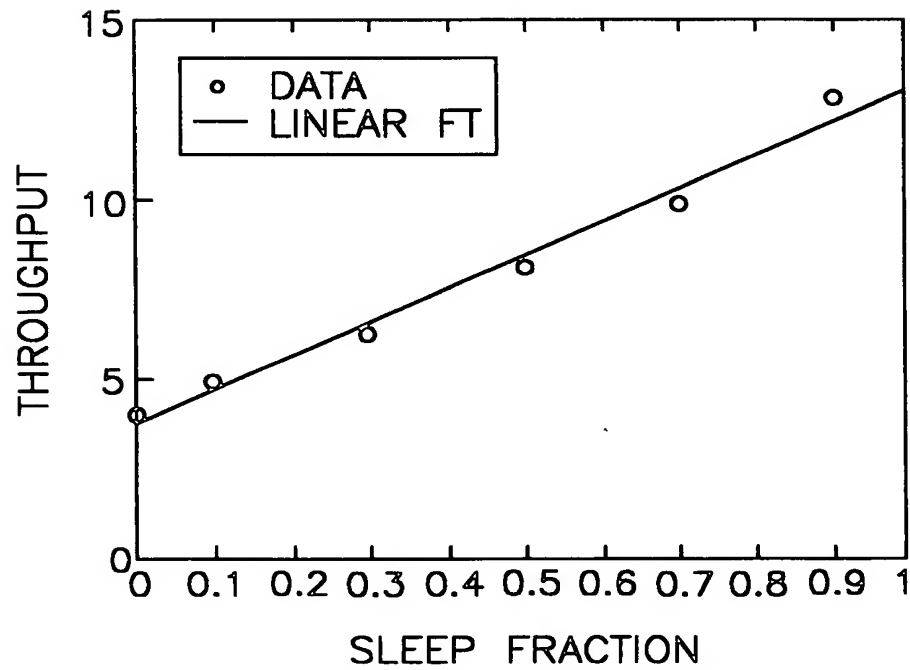


FIG. 12

(REPLACEMENT SHEET)
14/14
HURAS ET AL.
YOR920030458US1 (GHZ)

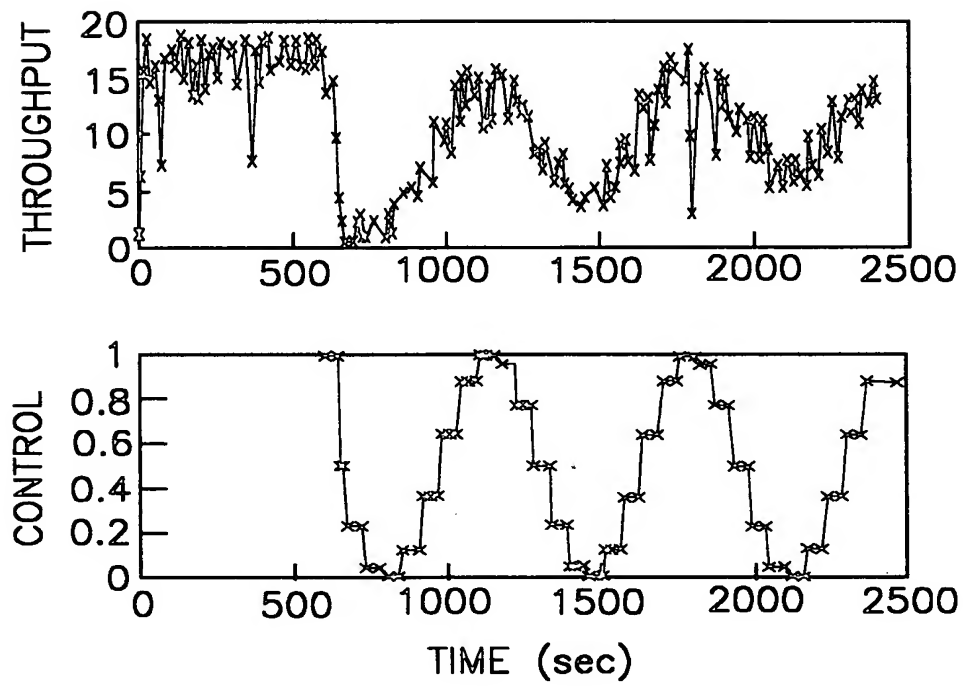


FIG. 13